

## From M-ary Query to Bit Query: a new strategy for efficient large-scale RFID identification

Article (Accepted Version)

Su, Jian, Chen, Yongrui, Sheng, Zhengguo, Huang, Zhong and Liu, Alex X (2020) From M-ary Query to Bit Query: a new strategy for efficient large-scale RFID identification. IEEE Transactions on Communications. p. 1. ISSN 0090-6778

This version is available from Sussex Research Online: <http://sro.sussex.ac.uk/id/eprint/89386/>

This document is made available in accordance with publisher policies and may differ from the published version or from the version of record. If you wish to cite this item you are advised to consult the publisher's version. Please see the URL above for details on accessing the published version.

### **Copyright and reuse:**

Sussex Research Online is a digital repository of the research output of the University.

Copyright and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable, the material made available in SRO has been checked for eligibility before being made available.

Copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

# From M-ary Query to Bit Query: a new strategy for efficient large-scale RFID identification

Jian Su, *Member, IEEE*, Yongrui Chen, *Member, IEEE*, Zhengguo Sheng, *Senior Member, IEEE*, Zhong Huang, Alex. X. Liu, *Fellow, IEEE*

**Abstract**—The tag collision avoidance has been viewed as one of the most important research problems in RFID communications and bit tracking technology has been widely embedded in query tree (QT) based algorithms to tackle such challenge. Existing solutions show further opportunity to greatly improve the reading performance because collision queries and empty queries are not fully explored. In this paper, a bit query (BQ) strategy based M-ary query tree protocol (BQMT) is presented, which can not only eliminate idle queries but also separate collided tags into many small subsets and make full use of the collided bits. To further optimize the reading performance, a modified dual prefixes matching (MDPM) mechanism is presented to allow multiple tags to respond in the same slot and thus significantly reduce the number of queries. Theoretical analysis and simulations are supplemented to validate the effectiveness of the proposed BQMT and MDPM, which outperform the existing QT-based algorithms. Also, the BQMT and MDPM can be combined to BQ-MDPM to improve the reading performance in system efficiency, total identification time, communication complexity and average energy cost.

**Index Terms**—RFID, anti-collision, bit query, bi-response, communication complexity.

## I. INTRODUCTION

AS one of the best known automatic identification technologies, RFID has attracted much attention for object management due to its low costs, fast recognition, reusability,

and robustness in harsh environments [1]. Currently, it has been widely used in commercial retail, traffic management, libraries, access control systems, food safety traceability, and other aspects [2-4]. The main components of an RFID system include a reader and several tags. Each tag maintains a unique identifier (ID) information that can be identified by the reader, which is the most common application paradigm of RFID. According to its power supply, the existing tag types include active, passive and semi-active. A passive tag does not have a built-in battery. When the tag is outside the radiation coverage of reader, it is in an inactive state. When it is within the reader vicinity, the tag extracts the power required for its operation from the RF energy emitted by the reader. The passive tag is characterized by low cost and long working lifetime, but reduced coverage than other tag types [5].

This paper mainly focuses on passive RFID systems especially consisting of a single reader and multiple tags. However, in such a system, multiple tags competing for a shared channel to transmit data at the same time leads to the tag collision problem [6], which not only hinders the successful transmission of tag data, but also increases time and energy consumption. Therefore, it is necessary to implement an anti-collision protocol to coordinate data communication between the reader and tags [7]. Due to hardware limitations of the RFID systems, general multiple access protocols are difficult to apply directly to the RFID system, thus there is a need to develop anti-collision protocols. The mainstream RFID anti-collision protocols can be divided into tree-based [8-14] and Aloha-based [15-20] protocols.

The Aloha-based protocols include Pure Aloha (PA) protocol, Slotted Aloha (SA) protocol, Frame Slotted Aloha (FSA) protocol, and Dynamic Frame Slotted Aloha protocol [21-22]. In Aloha-based protocols, tags pick up time slots randomly from a frame with a given size and respond to the reader at a designated slot. The feature of Aloha-based protocol is characterized by the strategy it employs to update the transmission frame size along the tag identification process. Such protocol is easy to implement, but its reading efficiency is usually low [23]. Moreover, all tags including those already being identified need to maintain their own inventory flags in Aloha-based protocols. Once some tags are powered down intermittently, they will contend with unidentified tags in the subsequent slots and cause high instability in RFID identification. In addition, the cardinality (the size of tag population) estimation functions in Aloha-based protocols not only increases the computational complexity, but also introduce estimation errors, which all cause performance degradation.

Manuscript received September 3, 2019; revised December 13, 2019; accepted January 14, 2020. This work is supported by the Natural Science Foundation of China (No.61802196); the Natural Science Foundation of Jiangsu Province (No.BK20180791); the Natural Science Foundation of Jiangsu Higher Education Institution of China (No.17KJB510036); and the Startup Foundation for Introducing Talent of NUIST. This work is partially supported by the National Science Foundation under Grant Number CNS-1837146. This work is also supported in part by Soft Science Program of China Meteorological Administration, Priority Academic Program Development of Jiangsu Higher Education Institutions, and Engineering Research Center of Digital Forensics, Ministry of Education. The associate editor coordinating the review of this article and approving it for publication was V. Aggarwal. (Corresponding authors: Yongrui Chen; Zhengguo Sheng.)

J. Su is with the School of Computer and Software, Nanjing University of Information Science and Technology, Jiangsu 210044, China (e-mail: sj890718@gmail.com).

Y. Chen is with the School of Electronic, Electrical, and Communication Engineering, University of Chinese Academy of Sciences, Beijing, 100190, P.R.China (e-mail: chenyr@ucas.ac.cn).

Z. Sheng is with the Department of Engineering and Design, University of Sussex, Brighton BN1 9RH, U.K. (e-mail: z.sheng@sussex.ac.uk).

Z. Huang is with the School of Information and Communications Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China. (Email: huangzhong\_2014@126.com)

A. X. Liu is with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824 USA (Email: alexliu@cse.msu.edu).

Digital Object Identifier xxxx

Tree-based protocols, in essence, split the competing tags into groups recursively until a successful response is detected by the reader. The tree-based protocols mainly include Query tree (QT) protocol [13-14][24-30], Binary tree (BT) protocol [8-12], and Binary search (BS) protocol [21]. Among them, QT protocol is the most common tree-based algorithm to improve the reading performance of RFID systems. In the QT protocol, each tag maintains a prefix matching circuit. A reader first probes prefix 0, and all tags whose IDs match the prefix 0 respond. If the feedback of the probe is a successful response (i.e., only one tag responds) or an empty response (i.e., no tag responds), the reader then probes prefix 1, and all tags whose IDs match the prefix 1 respond. If the feedback of the probe is a collided response (i.e., more than one tag respond), the reader updates two new prefixes by concatenating a 0 and a 1 after the previous prefix and probes the unread tags with the new prefixes. The reader maintains a stack to store the prefixes. When the stack is empty, it means that all tags are successfully identified and the identification process is terminated. The identification process of the QT can be regarded as a virtual depth-first traversal process on a complete binary tree, where each branch on the tree represents a certain bit of a tag ID, and a leaf node represents a successful response (named a readable slot) or an empty response (named an idle slot), and an internal node represents a collided response (named a collision slot) [25]. The main limitation of traditional QT protocol is that it traverses a large number of collision nodes in the binary tree, making the query process slow and thus increasing the identification latency. Although some methods utilize ad hoc heuristic features to reduce traversal of collision nodes, these methods cannot completely avoid it [24, 26].

Recently, many tag reading protocols have been proposed to enhance the reading performance of QT protocol by adopting the bit identification and tracking technology [14, 27] which enables the reader to accurately identify the location of collided bits in order to avoid continuous unnecessary collisions. Current research on the QT-based protocol focuses on using the collision information to update the query prefixes. Literature [28] presented a collision tree (CT) protocol which generates query prefixes and splits the contending tags according to the first collided bit. The authors in literature [29] proposed a protocol namely optimal query tracking tree (OQTT) which embeds a bit estimation algorithm (BEA) to allow reader to estimate the size of tag population. By estimating the tag population size, the reader divides the entire tag set into several groups and then utilizes the query tracking tree to quickly identify tags within the group. Such protocol greatly improves the system efficiency<sup>1</sup> of the original QT protocol, and its optimal system efficiency approaches 61.4%. However, both the BEA and partitioning strategy introduce additional costs such as counters, random number generators and storage memory, making them unsuitable for low-cost tags [14]. The literature [30] presented an adaptive assigned tree slotted Aloha (AdATSA) protocol, which is an improved version of ATSA [27]. In AdATSA, the reader employs

binary grouping and adaptive recognition phases to efficiently estimate the tag population size and promptly identify tags. The maximum system efficiency of AdATSA peaks at 61.7%. Another recent work namely dual prefix probe scheme (DPPS) [31] is proposed to enhance the system efficiency above 90%. However, these protocols only use the most significant bit (MSB) of the collided part, thus they may generate additional collision slots or empty slots. Although these prior work are more efficient than the original QT algorithm, there are still space to improve the time and energy efficiency.

To speed up the query process of QT, some M-ary based protocols [32-33] have been proposed to improve the identification performance. The literature [32] presented a M-ary query tree (MQT) by modeling the tag identification process as a partial deep-first traversal process on a virtual M-ary tree over the tag ID space. Specifically, the system efficiency of MQT is approaching 73%. In literature [33], a similar M-ary bit-detecting tree method (MCT) is proposed to rapidly identify the new arriving tags under a scenario in which the identified tags continue to participate in the identification process. Since the M-bit arbitration sequence needs to be fed back to the reader together with the tag ID, the communication overhead is increased.

The collision window tree (CwT) protocol has been proposed in [34]. In CwT, a window procedure is adopted to manage the tag response in order to reduce the energy wastage in collisions. However, such a protocol ignores the cost of head message in each query command. The head message is extremely important for communication between the reader and tags [13, 31]. Although CwT reduces the transmission overhead in a single time slot, it requires more time slots to identify the same batch of tags compared to other QT-based protocols, thus causing a degradation in system efficiency.

In this paper, we improve the identification efficiency of RFID system by looking into both query prefix update mechanism and tag response strategy. For query prefix update mechanism, we present a bit query (BQ) strategy based M-ary query tree protocol (BQMT). Specifically, after receiving a BQ message, the matching tags will respond to the reader with a mapped M-bit sequence indicating its location in the M-ary traversal tree. Compared with traditional ID query, it can both eliminate idle queries and separate collided tags into many small subsets which make full use of the collided bits. To further look at response strategy, we present a novel bi-response mechanism namely modified dual prefixes matching (MDPM) which allows multiple tags to respond in the same slot in order to reduce the total number of queries. Benefiting from bit query and bi-response mechanism, the combined BQ-MDPM can significantly improve the time and energy efficiency. Fig. 1 summarizes the evolution process and characteristics of the proposed solutions. The suitable scenarios for the proposed solutions are discussed in simulation section.

The reminder of the paper is organized as follows. Section 2 gives preliminary information, such as the Manchester coding, bit tracking technology and ID query. In Section 3, the transmission model between the reader and tags and the proposed BQMT is described in detail. In the following, the proposed MDPM and BQ-MDPM are described in Section 4,

<sup>1</sup> According to the definition of the existing literatures, the system efficiency is defined as the number of identified tags divided by the number of total slots.

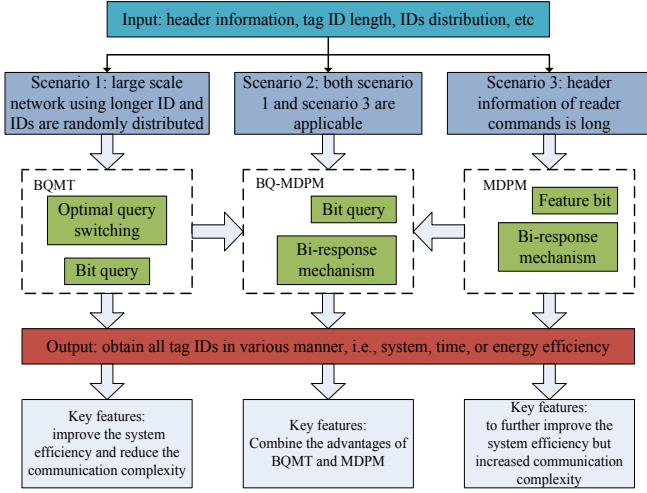


Fig. 1. Motivation and key features of the proposed solutions

followed by the performance evaluation results presented in Section 5. Finally, Section 6 draws conclusions.

## II. PRELIMINARIES: BIT TRACKING AND ID QUERY

In this paper, our RFID scenario consists of a portable reader and multiple passive tags, in which the reader does not have a priori knowledge of tag population or tag ID. We are looking at solutions to effectively collect all tag ID information when the reader read a given set of tags. Because tags communicate with the reader over a shared channel, such a challenge is also identified as tag identification problem. Since the tag population in a practical RFID system can be tens of thousands, the tag identification protocol should be scalable and efficient.

Bit tracking is commonly based on Manchester code [35-36], which defines the value of one bit as the voltage transition within a fixed time (called period). In RFID systems, tag transmission signal is based on the Manchester coding method. Therefore, as shown in Fig. 2, it is feasible to use bit tracking technology to trace collision bits. However, all tags within the reader vicinity should be synchronized to return data to the reader. Concerning the bit-wise synchronization of tag responses, the clocks of the tags can be well synchronized via the signal received from the reader [37] and small synchronization error does not matter since tags transmit at very low bit rates [38]. Such problem has been studied in the previous work [39-40] and is outside the scope of this paper. Moreover, the authors in the literature [41] recently verified that bit-wise synchronization is feasible in the UHF RFID system by carrying out the practical experiments with USRP and WISP tags. Also, we make similar assumption that a communication channel between the reader and tags is error-free and with no capture-effect [13-14][17-19][23-36].

Fig. 3 shows an example of QT protocol using bit tracking technology. The reader owns a stack  $S$  to record and update the query prefixes. The stack is initialized with an initial prefix (called empty string  $\varepsilon$ ). Each tag whose ID matches the query prefix will respond with its remaining ID (the ID excluding

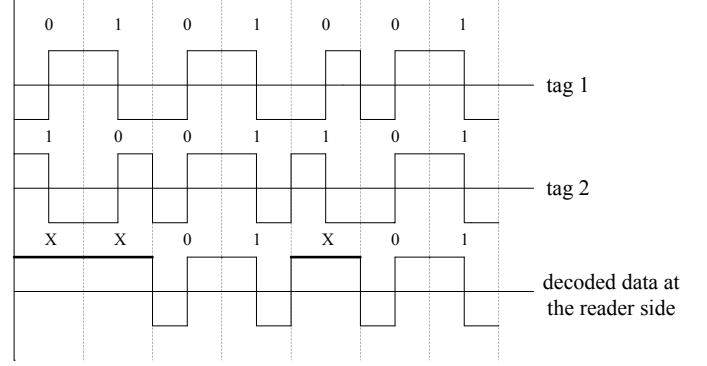


Fig. 2. Example of Manchester code

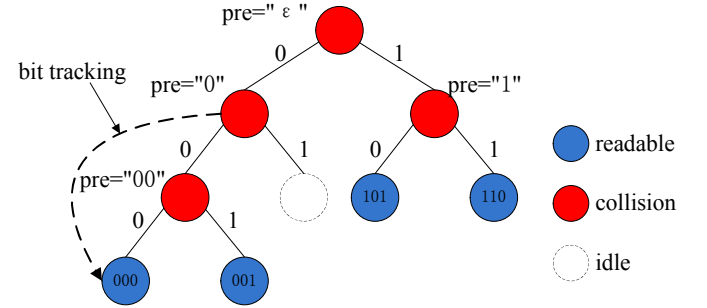


Fig. 3. Query tracking tree

same prefix to the reader). We name this query as ID query which is denoted briefly as  $iq$ . According to the responses of tags, a slot (node) has three possible status: collision, idle or readable. If more than one tags respond, the slot is a collision slot, the reader will update new queries with longer prefix and push them into the stack. If only one tag responds, the slot is readable, and the reader can identify the tag. If no tag responds, the slot is idle. Using the bit tracking technology, the reader can locate the first collision bit and skip unnecessary tree nodes. As observed from Fig. 3, we can know how many queries are required to identify a tag by using bit tracking technology. For example in this case, five slots are consumed to identify three tags.

## III. THE PROPOSED BQMT PROTOCOL

### A. System Model

The transmission model between the reader and tags follows the EPC C1 Gen2 standard. Fig. 4 illustrates the link timing of data exchange between the reader and tags for QT-based protocols. Similar to the Aloha-based protocols, in the identification process of QT based protocols, time is also divided into various number of slots. The reader initializes the identification process using Query command during time  $T_{Query}$ . After the first slot, the reader starts a slot using QR command during time  $T_{QR}$ . Throughout the whole tag identification process, the reader needs to transmit continuous wave (CW) to power the tags, allowing them to harvest energy and return their IDs. Therefore, how to shorten the identification time is a good way to improve energy efficiency. In Fig. 4,  $T_1$  is the time required for the tag to generate a response after each reader command.

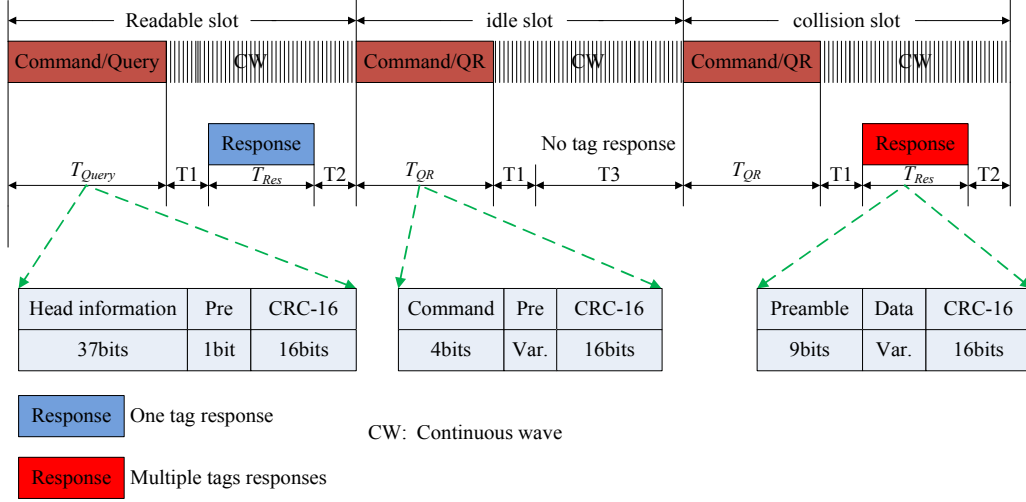


Fig. 4. The link timing between the reader and tags of QT-based protocols

$T_2$  is the time required for the reader to process the received tag response.  $T_3$  is the waiting time before the reader sending another command after  $T_1$  if there is no tag response.  $T_4$  is the guard time between two consecutive tag responses in the proposed MDPM. Additionally, the tag's response is produced during time  $T_{Res}$ .

During the tag identification process, a tag may be in one of three modes, that are waiting mode, sleeping mode or transmission mode. Specifically, if a tag is in waiting mode, it indicates that the tag's ID does not match the prefix in the command transmitted by the reader at the moment. Hence, the tag will not respond to the reader but will continue to wait for subsequent commands. When the ID of a tag matches the prefix sent by the reader, it is in the transmission mode and the tag will respond to the reader with its ID information.

### B. Bit Query

Based on the bit tracking technology introduced in the section II, we discuss the bit query method in details in this section. When the reader sends a bit query command, denoted as  $bq$ , the tag whose ID matches the prefix will respond to it. However, unlike other QT-based methods, the tag response to the command is no longer the remaining part of ID but a mapped  $M$ -bit string. The  $M$ -bit string is generated by the following  $m$  bits in the rest part of the ID (after the matched prefix), where  $M = 2^m$ . The mapped string is determined by the parameter  $m$  and the mapping function which is described in Algorithm 1, where the notation  $\ll$  means ROL operation. Tab. I shows the mapping table for  $m = 2$ . As observed in the Tab. I, the mapped data contains only a single 1. When the reader receives a mapped data, it can resolve the original information without additional time slots.

---

#### Algorithm 1 Mapping Function

---

**Input:**  $m$ -bit string  $b = b_{m-1}b_{m-2}...b_0$   
( $b_i$ : a binary value)

**Output:**  $M$ -bit string  $p = p_{M-1}p_{M-2}...p_0$   
( $M = 2^m$ ,  $p_i$ : a binary value)

---

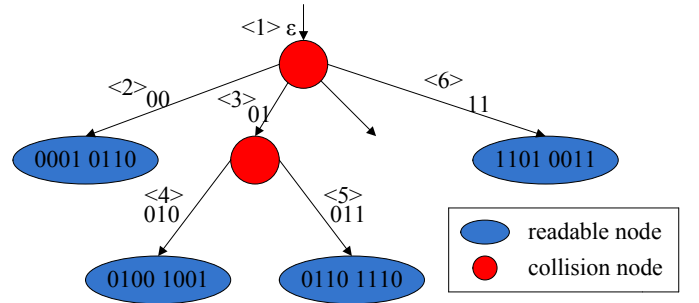


Fig. 5. M-ary query tree

- 1: Initialize  $p$  such that  $p_0 = 1$  and  $p_i = 0$  for all other  $i$
  - 2: **for**  $j = 0$  to  $m - 1$  **do**
  - 3:    $p \leftarrow p \ll b_j \times 2^j$
  - 4: **end for**
- 

TABLE I  
AN EXAMPLE OF THE MAPPING TABLE

Original Data ( $m = 2$ )	Mapped data ( $M = 2^m = 4$ )
00	0001
01	0010
10	0100
11	1000

TABLE II  
AN EXAMPLE OF RECEIVED STRING

Original prefix	Mapped string
Tag A (00)	0001
Tag B and Tag C (01)	0010
Tag D (11)	1000
Received string at the reader side	*0**

An identification process of four tags represented by an M-ary query tree is shown in Fig. 5. The number in the angle

brackets indicates the index value of the slot. The number on the branch denotes the query prefix sent by the reader at this slot. When the reader initially broadcasts a *bq* command with the prefix  $\varepsilon$ , all tags that match the prefix will map the upper 2 bits of its ID to a 4-bit data string according to the mapping principle described in Algorithm 1, and respond it to the reader. Based on the received signal at the reader side indicated in Tab. II, the reader parses out that “10” branch on the query tree is idle. That is to say, if the reader sends a query command with a prefix of “10”, it will not receive any tag response. Since “00”, “01”, and “11” are non-idle nodes, the reader uses traditional ID query (denoted as *iq*) to identify the individual tags. When the reader broadcasts a *iq* command with the prefix of “00” or “11”, it can obtain a single ID with a correct cyclic redundancy check (CRC) checksum. When the reader sends a *iq* command with the prefix of “01”, two tags compete for a shared channel at the same time, causing ID sequence collided. Then the reader has to query them again until they are successfully identified.

For bit query, the length ( $M$ ) of mapped bit string needs to be determined. Intuitively, the larger the  $M$  value, the more subsets can be separated from a collision slot, so that fewer number of queries can be achieved. However,  $M$  should not be greater than the length of ID sequence if the slot duration is not elongated. Typically, the length of ID is set to 128 bits as in [13-14, 36]. Based on this assumption, we set  $M = 128$ , i.e.,  $m = 7$  to follow the existing settings. It is noted that the larger the  $M$  value, the better performance of the proposed algorithm can be achieved. But this characteristic does not apply to other comparative algorithms. For example, in MCT algorithm, the number of idle slots increases with the number of subsets, and thus leading to increased time and energy costs. Thus, even if MCT algorithm can divide the colliding tags into more subsets like BQMT, it is unable to achieve better performance. Compared to other algorithms, the proposed BQMT can completely remove idle slots and improve the identification efficiency.

### C. Optimal Query Switching

Although bit query is better than ID query in resolving collisions, it also has many shortcomings. For example, suppose a node in the query tree is readable. In other words, if the node contains only one tag, then ID query can be used to obtain the complete ID of the tag, while the bit query can be used to only obtain the  $m$  bits of the ID. Therefore, to achieve efficient tag identification, an optimal query switching strategy should be determined. Consider an ideal situation where the reader knows in advance that the next time slot has only a single tag response, then it can switch the query mode from bit query to ID query. However, since a collision bit in a received string can only reflect the fact that the corresponding child node is a non-idle node, the reader cannot accurately predict how many tag responses will be made in the next query cycle. Thus, to find the optimal query switching strategy, the reader has to estimate the number of tag in a non-idle node. The estimation method is as follows.

Assuming that the reader receives the responding strings from  $n$  tags at the same time, in the mixed string the number of

“0” is  $E$ , and the number of collision bits is  $C$  ( $E + C = M$ ). Assuming that the IDs of tags are uniformly distributed, the probability of finding  $k$  tags in a child node can be expressed as

$$B(k) = \binom{n}{k} \left(\frac{1}{M}\right)^k \left(1 - \frac{1}{M}\right)^{n-k} \quad (1)$$

where  $n$  is the number of tags waiting to be identified,  $M$  is the length of mapped string received by the reader. Then the expectation of number of idle nodes can be calculated as

$$E = M \times B(0) = M \times \left(1 - \frac{1}{M}\right)^n \quad (2)$$

When the value of  $M$  is large enough, we have

$$\left(1 - \frac{1}{M}\right) \approx e^{-\frac{1}{M}} \quad (3)$$

Accordingly, the (3) can be further approximated as

$$E \approx M \times e^{-\frac{n}{M}} \approx M e^{-\lambda} \quad (4)$$

where  $\lambda = \frac{n}{M}$ .

For a parent node  $p$ , the number of non-idle child nodes subordinate to it is  $C$ , and the number of tags contained in each child node  $c_i$  is  $k_i$  ( $i = 1, \dots, C$ ). If an ID query is used to access a child node, according to the existing literature [28, 31, 36], the number of time slots required to successfully identify all tags contained in the child node can be calculated as

$$N_{iq}(c_i) = 2 \times k_i - 1 \quad (5)$$

Therefore, the number of slots to identify all tags belong to the parent node  $p$  can be calculated as

$$\begin{aligned} N_{iq}(p) &= 1 + \sum_{i=1}^C N_s(c_i) = 1 + \sum_{i=1}^C (2 \times k_i - 1) \\ &= 2n - C + 1 \end{aligned} \quad (6)$$

On the other hand, if we use the bit query method to visit the child nodes, we can first make an assumption that most child nodes contain only a small number of tags (far less than  $M$ ). Such assumption is reasonable because the previous analysis shows that we only need to switch the query when the number of tags contained in each child node is small. Otherwise, if there are more tags, the bit query is significantly more efficient than the ID query. Thus, for these child nodes, if they are visited by bit queries, most of grand-children are more likely to be readable nodes, which means that only one slot is needed to identify one of them. Therefore, using a bit query to identify a child node  $c_i$  containing  $k_i$  tags, the number of slots required can be calculated as

$$N_{bq}(c_i) = k_i + 1 \quad (7)$$

where  $k_i$  represents the number of grand-child nodes, and 1 represents current child node. Accordingly, the total number of slots to identify all tags contained in the parent node  $p$  via bit query can be expressed as

$$\begin{aligned} N_{bq}(p) &= 1 + \sum_{i=1}^C N_s(c_i) = 1 + \sum_{i=1}^C (k_i + 1) \\ &= n + C + 1 \end{aligned} \quad (8)$$

If ID query is more efficient than bit query, the following

is satisfied.

$$N_{bq}(p) \geq N_{iq}(p) \quad (9)$$

Based on (6) and (8), we have

$$C \geq \frac{1}{2}n \quad (10)$$

According to (4) and  $E + C = M$ ,  $C$  can be further expressed as

$$C = M - E = M(1 - B(0)) \approx M(1 - e^{-\lambda}) \quad (11)$$

Substituting (11) into (10), we have

$$\begin{aligned} M(1 - e^{-\lambda}) &\geq \frac{1}{2}\lambda M \\ \Rightarrow \frac{1 - e^{-\lambda}}{\lambda} &\geq \frac{1}{2} \\ \Rightarrow \lambda &\leq 1.6 \end{aligned} \quad (12)$$

Based on (4) and (12), we have

$$\begin{aligned} E &\approx M e^{-\lambda} \geq 0.2M \\ \Rightarrow \frac{E}{M} &\geq 0.2 \end{aligned} \quad (13)$$

The (13) indicates that when the query should be switched. That is, if the ratio  $E/M$  of a received string is higher than 0.2 it is more efficient to switch to ID query.

#### D. Performance Analysis of BQMT

To derive the performance of the proposed BQMT, let us assume that after the reader receives the mapped string from tag feedback, it can accurately predict the number of tags corresponding to each collision bit (child node), then it can perfectly switch to the optimal query approach. If a child node is a collision node, the reader will send a bit query to probe this node. If a child node is a readable node, the reader will send a ID query to probe this node. Otherwise, the reader will skip the idle node. It is noted that an initial query is bit query, we can derive the total number of queries for the reader to identify  $n$  tags as

$$N_q = \begin{cases} 2, & n = 1 \\ 1 + M \times B(1) \\ + \sum_{k=2}^n M \times B(k) \times N_q(k), & n > 1 \end{cases} \quad (14)$$

where  $B(k)$  is the probability of finding  $k$  tags in a child node, and is given by (1). Although it is difficult to derive a closed form solution since  $B(k)$  is different for each individual  $n$ , we can obtain an upper bound of  $N_q(n)$  as follows.

**Lemma 1.**  $N_q(n)$  has a upper bound, that is

$$N_q(n) \leq 1.5n + 1 \quad (15)$$

*Proof:* For the root node in the M-ary query tree, if the node is probed by ID query, the number of slots consumed by BQMT is expressed as (6). On the other hand, if the node is probed by bit query, the number of slots consumed by BQMT is expressed as (7). Therefore, assuming that BQMT optimally choose one query, we have

$$\begin{aligned} N_q(n) &= \min(N_{iq}(p), N_{bq}(p)) \\ &= \min(2n - C + 1, n + C + 1) \\ &\leq 1.5n + 1 \end{aligned} \quad (16)$$

The equal sign of the above formula holds when  $C = 0.5n$ , and the lemma 1 can be yielded. ■

**Lemma 2.** Assume the number of tags to be identified is  $n$ , the minimal system efficiency of BQMT is 0.66.

*Proof:* According to the lemma 1, the maximum number of slots consumed by BQMT to identify  $n$  tags can be expressed as  $N_q^{\min}(n) = 1.5n + 1$ . Hence, the minimum system efficiency can be derived as

$$U_{\min}^{BQMT} = \frac{n}{1.5n + 1} \geq 0.66 \quad (17)$$

#### IV. AN IMPROVED READING PROTOCOLS NAMELY MDPM AND BQ-MDPM

From the perspective of prefix update, we design a bit query strategy and BQMT protocol in section 3. However, the performance of BQMT is affected by the number of subsets, i.e., the value of  $M$ . And there is still space for improvement in system efficiency. In this section, we design a modified dual prefixes matching (MDPM) mechanism to allow more than one tags respond to the reader in a same slot illustrated in Fig. 6 and hence further reduce the total number of queries. The BQ-MDPM is designed to improve the reading performance in system, time and energy efficiency.

##### A. Definitions

Before describing the MDPM in detail, we first introduce several definitions.

- **Feature bit:** In the proposed solution, the feature bit is the MSB of the remaining ID after the tag matches the prefix of the reader command. The tag can use this feature bit to determine which sub-cycle it belongs to and respond to the reader. Given a binary string  $P_1P_2...P_k$  as a query prefix and  $R_1R_2...R_L$  as full ID of a tag (where  $k \leq L$ ), if  $P_1P_2...P_k = R_1R_2...R_k$ , then  $R_{k+1}$  is the feature bit. Obviously, the specific position of the feature bit is determined by the tag ID waiting to be identified, so it is dynamically changed during the tag identification process. Specifically, if  $k = L$ , the reader can identify the tags according to the duality of a binary value.
- **Sub-cycle:** In QT-based protocols, one cycle (include reader query and tag response) can be equivalent to a time slot in the Aloha-based protocols. To be specific, a cycle is the time duration from the reader sending a query command to successfully receiving the tag response, which is illustrated in Fig. 3. Unlike the traditional QT-based protocols, the proposed MDPM uses bi-response mechanism to identify two tags in a slot. Thus, in this paper, a cycle is divided into two sub-cycles: sub-cycle A and sub-cycle B. It is noted that the reader only needs to send a query command in each cycle. The query command contains three parameters: Com\_Str, pre1 and pre2. A tag that receives the query command will determine which sub-cycle it belongs to according to whether its own feature bit matches pre1 or pre2.



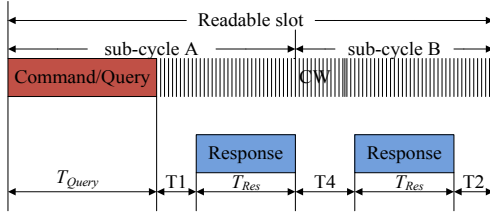


Fig. 6. The link timing of a readable slot transmission in MDPM

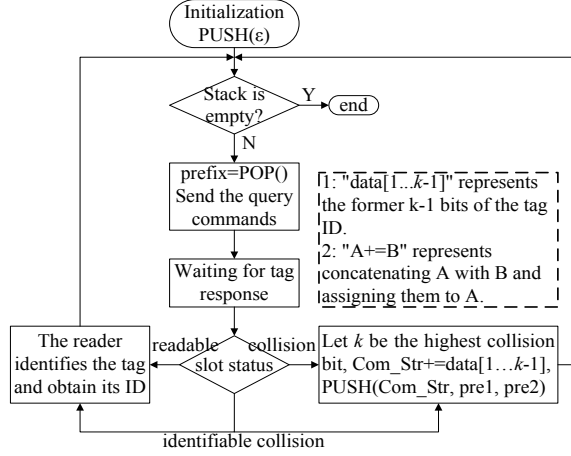


Fig. 7. Flowchart of MDPM for reader

- **Bi-response mechanism:** As described above, the feature bit determines which sub-cycle the tag selects to respond to the reader. The so-called bi-response mechanism is that tags from different subsets may respond to the reader's query commands in the same cycle (but in different sub-cycles). The tags whose feature bit equals to 0 will firstly respond to the reader in the sub-cycle A. Then those tags with feature bit of 1 respond to the reader in the sub-cycle B. Specifically, the tags in sub-cycle B respond to the reader after a  $r$ -bits time delay, where  $r = l(ID - Com\_Str - pre1)$ , in which  $l$  represents calculating the length of binary string.

### B. MDPM: Algorithm Description

The detailed flowchart of the proposed MDPM is described in Fig. 7. Strictly speaking, the proposed MDPM is a new kind of QT-based algorithm. So, communication flow of the proposed MDPM is similar to that of the traditional QT-algorithm. The detailed difference between them is as follows.

- 1) **Prefix updating mechanism:** In traditional QT-algorithms, the query prefixes are only based on the previous query prefixes, which cause many unnecessary empty slots. As a contrary, the query prefixes of MDPM are based on both the previous prefixes and the received bit string, and hence significantly avoid the unnecessary empty slots.
- 2) **Tag response:** In traditional QT-algorithms, the tag responds to the reader with its full ID or partial ID excluding the matched prefix. Only one tag is allowed to transmit ID in a slot. As a contrary, the proposed

(a)	Header information	Pre	Pre1	Pre2	CRC-16
	37bits	1 bit	0	1	16 bits

(b)	Preamble	ID	CRC-16
	9 bits	128 bits	16 bits

(c)	Command	Com_Str	Pre1	Pre2	CRC-16
	4 bits	Var.	0	1	16 bits

(d)	Preamble	Data	CRC-16
	9 bits	Var.	16 bits

Fig. 8. The format of the reader's probe commands and the tag's response (a) CMD\_Query command (b) tag's response for CMD\_Query command (c) CMD\_QR command (d) tag's response for CMD\_QR command

solution allows two sets of tags with different feature bits responding to the reader in two sub-cycles. In this way, the reader can identify up to two tags in a time slot.

- 3) **Query command parameters:** In traditional QT-algorithms, the query command contains only one prefix used to allow the matching tag respond to its ID. As a contrary, the query command in the proposed method contains three parameters: Com\_Str, pre1, and pre2, where the function of Com\_Str is the same as the prefix in the traditional QT-algorithm. pre1 and pre2 are used to achieve the bi-response mechanism.

Fig. 8 gives the detailed command format and the corresponding tag response, where header information includes the header, mask, command and address information. Note that pre1 and pre2 are fixed to 0 and 1, respectively. They are used to match the feature bits of tags.

Tab. III gives an example of communication procedure by using MDPM to identify four tags whose IDs are "00010110", "01001001", "01101110", and "11010011". The reader initially sends a CMD\_Query command with an empty string of  $\varepsilon$  to start the identification process. In each slot, two sets of tags with different feature bits respond to the reader's query commands in two sub-cycles: sub-cycle A and sub-cycle B, respectively. It is noted in Fig. 5 and Tab. III, there are three kinds of slots in our proposed MDPM.

- **Readable slot:** If the tags are successfully identified in two sub-cycles in current slot, then the slot is named as a readable slot. In the example as described in Tab. III, the slot <3> is a readable slot. In this time slot the reader identifies two tags, B and C, respectively.
- **Collision slot:** If collisions occur in both of two sub-cycles, the reader cannot identify any tag in the slot, then it is called a collision slot.
- **Identifiable collision slot:** The reader can successfully identify a single tag in one of two sub-cycles, then the slot is an identifiable collision slot. For example, in Tab. III, slot <1> and slot <2> are identifiable collision slots.

As can be found in Tab. III, the proposed solution only needs three slots to identify four tags, thus greatly reducing



TABLE III  
COMMUNICATION PROCEDURE FOR AN EXAMPLE BY USING MDPM

Slot	Query (COM_Str, pre1, pre2)	Response		Identification
<1>	CMD_Query ( $\varepsilon$ , 0, 1)	0*****	11010011	Tag D is identified
<2>	CMD_QR (0, 0, 1)	0010110	1*01***	Tag A is identified
<3>	CMD_QR (01, 0, 1)	001001	101110	Tags B and C are identified

the number of queries from the reader. Although the duration of a single time slot is increased, as the total number of time slots is reduced and coordinate communication time can be potentially reduced, thereby improving the overall identification efficiency.

### C. BQ-MDPM: Algorithm Description

Although the performance of BQMT and MDPM has been greatly improved compared to prior arts, they still have many limitations. For BQMT, its performance highly depends on the  $M$  value, which is related to the length of tag ID. Therefore, in the scenario where the length of tag ID is short, its performance will deteriorate sharply. For MDPM, Although it can greatly reduce the total number of queries, it also brings some disadvantages. In particular, the MDPM has no advantages in communication complexity, identification time and energy cost especially under the scenario that the header information of reader commands can be ignored [28][34]. Because it allows more than one tags to respond in a same slot, thereby extending the duration of a single time slot. To overcome the limitations of BQMT and MDPM, the combined BQ-MDPM can be proposed to mitigate shortcomings in certain scenarios. The BQ-MDPM is described as follows. The reader broadcasts a  $bq$  query to initialize the identification process. After receiving the  $bq$  query, all tags respond to reader with a mapped string. The reader updates the prefixes according to the received string and switches the following queries according to Algorithm 2. If the following query is still a bit query, the above process continues. If the following query is an ID query, the reader allows tags to respond to the query according to the bi-response mechanism and their feature bits. Tab. IV gives an example of communications procedure by using BQ-MDPM to identify same four tags described in Tab. III. As can be observed, although BQ-MDPM consumes more slots than MDPM, it requires lower communication complexity than MDPM.

## V. NUMERICAL RESULTS

### A. Simulation Setup

In this section, we compared the proposed BQMT and MDPM with existing state-of-the-art solutions over extensive Monte Carlo simulations. Simulation scenarios with a reader and a various number of tags have been evaluated using MATLAB R2012b, where the tags are uniformly distributed in the reader vicinity so that all tags can receive the reader's command with no errors. The communication channel between the reader and tags are considered as ideal, as same as that in the previous literatures [13-14][23-28][31-36]. The parameters

TABLE V  
THE PARAMETERS USED IN THE MATLAB SIMULATIONS [13][24][34]

Parameter	Value
$L$	128 bits
Data rate	160 kbps
T1	25 $\mu$ s
T2	25 $\mu$ s
T3	30 $\mu$ s
T4	12.5 $\mu$ s
$P_{tx}$	825 mw
$P_{rx}$	125 mw
The number of simulations	1000

used in MATLAB simulations are listed in Tab. V. In our simulations, the tag number is from 100 to 2000. All the simulation results that we report in this paper were performed in the Lenovo desktop with Intel i5-4590 CPU and 8GB RAM. In the simulations, the performance is averaged over 1000 iterations.

### B. Results on Numerous Metrics

1) *Number of total slots:* The BQMT reduces the number of total slots of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 48.5% and 32.5%, respectively for uniformly distributed tag populations. Further, the MDPM reduces the number of total slots of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 56.9% and 43.5%, respectively. Fig. 9 (a) compares the number of total slots used by various algorithms where the tag population is from 100 to 2000. As can be observed from Fig. 9 (a), MDPM consumes the least slots to identify the same number of tags compared to other algorithms. The reason is that the introduced Bi-response mechanism can significantly reduce the number of collision slots and remove the idle slots, and thus save the total number of slots. Also, the MCT algorithm outperforms CwT and STT because M-ary tree can divide the colliding tags into smaller subsets and avoid the idle query nodes.

2) *System efficiency:* According to the definition of the existing literatures [16-20], the system efficiency is computed as the number of identified tags divided by the number of total slots to identify these tags. Such metric is widely used to evaluate the performance of RFID anti-collision protocols. The BQMT outperforms all other reference methods and improves the normalized system efficiency of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by

TABLE IV  
COMMUNICATION PROCEDURE FOR AN EXAMPLE BY USING BQ-MDPM

Slot	Query (Com_Str, pre1, pre2)	Response		Identification
<1>	CMD_Query ( $\varepsilon, \varepsilon, \varepsilon$ )	x0xx		
<2>	CMD_QR (00, 0, 1)	010110		Tag A is identified
<3>	CMD_QR (01, 0, 1)	001001	101110	Tags B and C are identified
<4>	CMD_QR (11, 0, 1)	010011		Tag D is identified

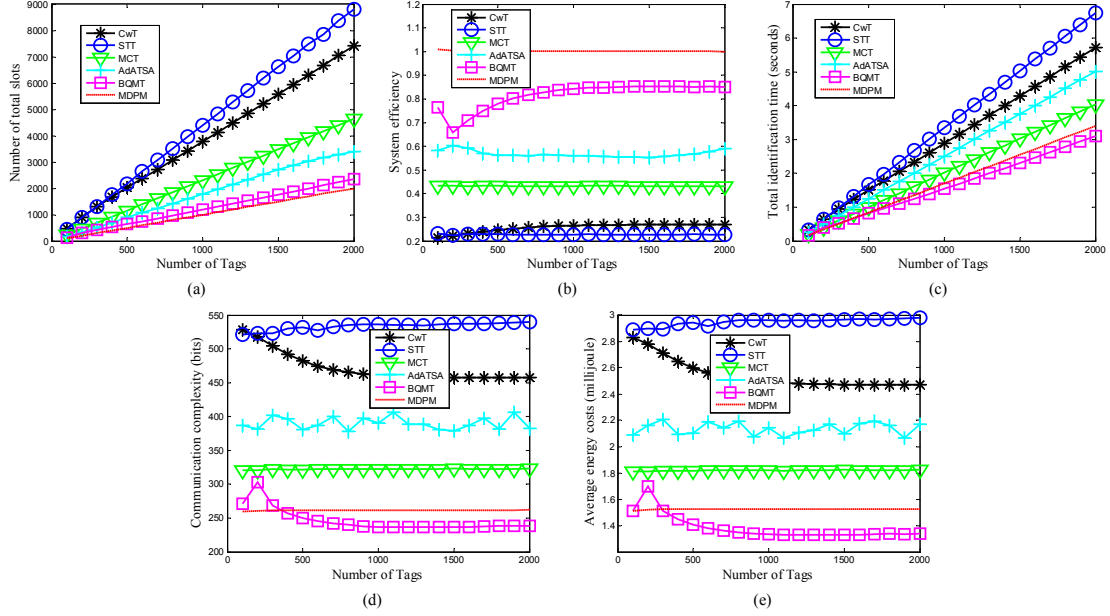


Fig. 9. Comparison of various algorithms: (a) Number of total slots (b) System efficiency (c) Total identification time (d) Communication complexity (e) Average energy cost for one tag identification

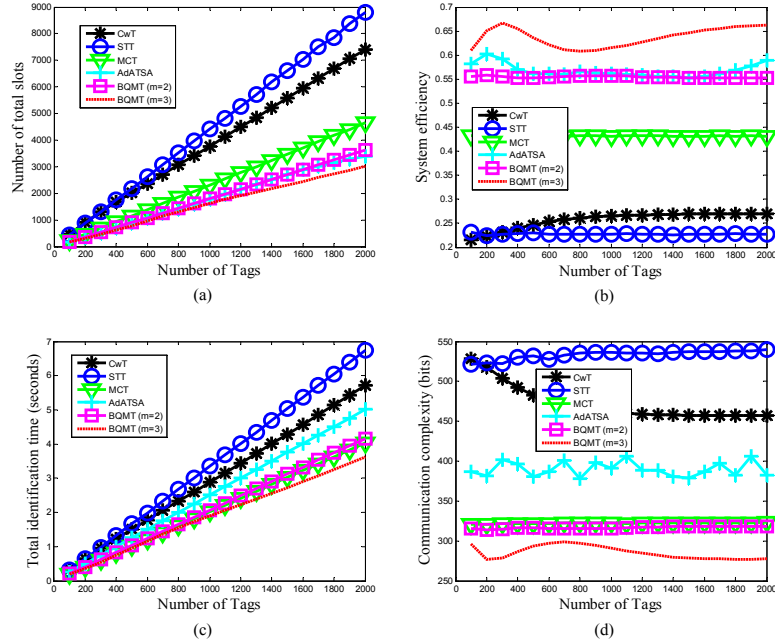


Fig. 10. Performance comparison between various algorithm and BMQT with  $m = 2$  and  $m = 3$ : (a) Number of total slots (b) System efficiency (c) Total identification time (d) Communication complexity

an average of 88.6% and 43.6%, respectively for uniformly distributed tag populations. Further, the MDPM improves the system efficiency of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 132% and 76.7%, respectively. Fig. 9 (b) compares the system efficiency of all comparative approaches. Similar to the results in Fig. 9 (a), BQMT always achieves the better performance than other comparative algorithms, and the system efficiency is close to 0.89 when the number of tags is greater than 800. The reason is that compared with other schemes, the BQMT can separate the contending tag set into more small subsets which are easier to be identified. Thus, the number of collision queries are greatly reduced and the system efficiency is improved. Since the CwT attempts to reduce the number bits transmitted in each query cycle by using the window structure, it requires to increase the total number of queries, which dramatically reduces the system efficiency. Although MCT and AdTASA utilize bit tracking technology which can avoid idle slots when resolving collision, it is unable to eliminate the idle slots during the initial phase of tag identification which degrade the system efficiency. The STT algorithm increases the number of idle queries while reducing the collisions, resulting in increasing in total number of queries, thus reducing the system efficiency. Benefiting from the bi-response mechanism, the system efficiency of the proposed MDPM is above 1.

3) *Total identification time*: As mentioned in the previous work [7][11][19][24][34], the system efficiency is ineffective to evaluate the actual performance of anti-collision approaches because it assumes the time interval of different types of slots are equal. In addition, the number of time slots varies from algorithm to algorithm. *The proposed BQMT reduces the total identification time of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 22.7% and 37.8%, respectively. Further, the MDPM reduce the total identification time of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 15.7% and 32.2%, respectively.* Fig. 9 (c) depicts the total identification time required for various algorithms to identify the same number of tags. The total identification time highly depends on the time duration for data transmission and the number of queries. It can be observed that different algorithms exhibit different ranks under different performance evaluation metrics. For example, although the MCT algorithm consumes more number of queries than AdTASA, the total identification time required is shorter. That is because in the MCT algorithm, both the reader and tags need to send less number of transmitted bits, thereby shortening the total identification time. Also can be observed from Fig. 9 (c), although MDPM reduces the total number of queries, it requires longer identification time than BQMT to identify the same number of tags due to it doubles the duration in each slot. Therefore, although MDPM brings the improvement of system efficiency, it does not bring further advantages in total identification time.

4) *Communication complexity and average energy cost*: The communication complexity is defined as the average number of bits transmitted by the reader and tags during RFID identification process, which implicitly represent the energy cost during the communication process [7][13][24][42][43].

*The BQMT reduces the average number of transmitted bits per tag identification of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 23.6% and 36.8%, respectively. Further, the MDPM reduces the average number of transmitted bits for one tag identification of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 34.3% and 45.7%, respectively.* Fig. 9 (d) demonstrates the average number of transmitted bits per tag identification for all comparative algorithms. Although STT is used to reduce the number of collisions in original QT method using a tree traversal path, it generates successive number of queries and increases the communication complexity. In addition, the STT requires full ID response to the reader, thereby increasing the transmission overhead. Therefore, the STT consumes more transmitted bits to identify one tag compared to other algorithms. It is noted that the MCT requires least number of transmitted bits among the comparative QT-based algorithms. The main reason is two-fold. First of all, the MCT consumes fewer number of queries by constructing a new M-ary collision tree structure. More importantly, besides the first slot of a frame, other slots are triggered by short commands, thereby reducing the transmitted bits by the reader.

*The proposed BQMT reduces the average energy cost per tag identification of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 23.9% and 35.3%, respectively. Further, the MDPM reduces the average energy cost per tag identification of the best prior passive RFID compliant QT-based and hybrid anti-collision protocols by an average of 30.3% and 40.7%, respectively.* Fig. 9 (e) compares the average energy cost for one tag identification. It also shows evidence that BQMT and MDPM outperform the other comparative algorithms in terms of energy consumption. This is because BQMT and MDPM produces fewer number of slots and transmitted bits at both the reader and tags sides.

5) *Impact of  $M$  value on BQMT performance*: In the BQMT design, the reader requires tags to respond a mapped data by sending a bit query command. Such mapped data can be potentially used to divide the colliding tags into  $M$  ( $M = 2^m$ ) groups, where  $M$  denotes the length of mapped data. To allow successful transmission of  $M$  bits data in a slot, we limit value  $M$  no more than the length of tag ID. That is,  $m = 7$  in the proposed BQMT. The larger the  $M$  value, the better performance of the proposed BQMT can be achieved. But we agree that this characteristic does not apply to other comparative algorithms. For example, in the MCT algorithm, the number of idle slots increases with the number of subsets, thus it leads to increased time and energy costs. Therefore, even if the MCT algorithm can divide colliding tags into more subsets like BQMT, it is unable to achieve better performance. Compared to other algorithms, the proposed BQMT can completely remove idle slots and improve the identification efficiency.

To better illustrate the advantage of our proposed solution, we provide the numerical results of the proposed BQMT in the cases of  $M = 4$  ( $m = 2$ ) and  $M = 8$  ( $m = 3$ ), respectively. According to the results illustrated in Fig. 10, we know that

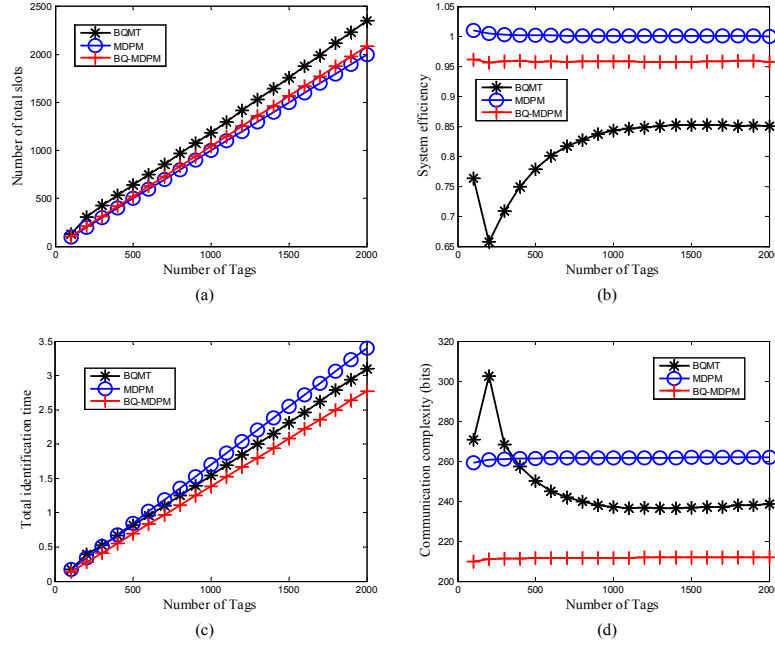


Fig. 11. Performance comparison between the proposed three solutions: (a) Number of total slots (b) System efficiency (c) Total identification time (d) Communication complexity

the performance of  $m = 2$  and  $m = 3$  are still superior to other QT-based protocols.

### C. Comparison of BQMT, MDPM and BQ-MDPM

In this section, we verify the effectiveness of BQ-MDPM by comparing its performance with BQMT and MDPM. Fig. 11 (a) plots the number of total slots for all comparative algorithms. As can be observed, the BQ-MDPM reduces the number of total slots of BQMT by an average of 12.6%. However, compared to MDPM, BQ-MDPM requires more slots to identify all tags due to it needs to send extra  $bq$  queries. Fig. 11 (b) compares the system efficiency of the proposed three protocols. Benefiting from the bi-response mechanism, BQ-MDPM outperforms BQMT by an average of 17.7%. Similarly in Fig. 10 (b), BQ-MDPM is slightly worse than MDPM in system efficiency. Fig. 11 (c) depicts the total identification time required for various algorithms to identify the same number of tags. As can be observed, BQ-MDPM consumes the least time compared to BQMT and MDPM, and reduces the total identification time by an average of 11.0% and 18.4%, respectively. As mentioned above, the total identification time highly depends on the time duration for data transmission and number of queries. The BQ-MDPM compromises between number of queries and transmitted bits. Fig. 11 (d) demonstrates communication complexity (which is defined as the average number of bits transmitted by the reader and tags during RFID identification process) for all comparative algorithms. The BQ-MDPM reduces the communication complexity of BQMT and MDPM by an average of 14.2% and 19.5%, respectively. The results also show evidence that BQ-MDPM outperforms BQMT and MDPM in communication complexity. In summary, BQ-MDPM inherits the characteristics of BQMT and MDPM, so it can show a

balance performance under various performance evaluation metrics. From the principle and numerical results of the proposed solutions, we can obtain the following conclusions. The BQMT is suitable for large scale network using longer IDs. Where the tag IDs are randomly distributed. The MDPM is suitable for the scenario where the header information of the reader command is long. No matter what scenarios, the proposed BQ-MDPM is a tradeoff between BQMT and MDPM.

## VI. CONCLUSION

In this paper, we have presented three time and energy-efficient protocols for RFID tag identification, namely the BQMT, MDPM, and BQ-MDPM respectively. In BQMT, two query types are utilized, ID query and Bit query. In bit query, an original  $m$  bits of a tag can be mapped into a  $M$  bits string to form a  $M$ -ary tree. According to the designed switching mechanism, the reader can optimally choose the query type for the ongoing query. To further optimize the reading performance of QT-based anti-collision protocols, a MDPM protocol is proposed to speed up the tag identification process and reduce the energy consumption during identification process. As a combination of BQMT and MDPM, the BQ-MDPM inherits the advantages of them and can achieve the better performance depends on application scenarios. Both theoretical analysis and numerical results have shown that the proposed protocols significantly outperforms all prior tag identification protocols including QT-compliant and Hybrid, for various evaluation metrics such as the number of total slots, system efficiency, total identification time, communication complexity, and average energy cost.

## REFERENCES

- [1] A. Lehto, J. Nummela, L. Ukkonen, L. Sydanheimo, and M. Kivikoski, "Passive UHF RFID in paper industry: challenge, benefits and the application environment," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 66-79, 2008.
- [2] F. Gandino, B. Montrucchio, M. Rebaudengo, and E. R. Sanchez, "On improving automation by integrating RFID in the traceability management of the Agri-Food sector," *IEEE Trans. Ind. Electron.*, vol. 56, no. 7, pp. 2357-2365, 2009.
- [3] V. D. Hunt, A. Puglia, M. Puglia, RFID: a guide to radio frequency identification, *John Wiley and Sons*, 2007.
- [4] M. Bolic, D. Somplot-Ryl, and I. Stojmenovic, RFID systems: research trends and challenges, *John Wiley and Sons*, 2010.
- [5] D. K. Klair, K.-W. Chin, and R. Raad, "A survey and tutorial of RFID anti-collision protocols," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 3, pp. 400-421, 2010.
- [6] D. Shih, P. Sun, D. Yen, and S. Huang, "Taxonomy and survey of RFID anti-collision protocols," *Comput. Commun.*, vol. 29, no. 11, pp. 2150-2166, 2006.
- [7] J. Su, Z. Sheng, V. Leung, and Y. Chen, "Energy efficient tag identification algorithms for RFID: survey, motivation and new design," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 118-124, 2019.
- [8] J. Myung, W. Lee, J. Srivastava, and T. K. Shin, "Tag-splitting: adaptive collision arbitration protocols for RFID tag identification," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 6, pp. 763-775, 2007.
- [9] Y. Cui and Y. Zhao, "Performance evaluation of a multi-branch tree algorithm in RFID," *IEEE Trans. Commun.*, vol. 58, no. 5, pp. 1356-1364, 2010.
- [10] J. S. Li and Y.-M. Huo, "An efficient time-bound collision prevention scheme for RFID re-entering tags," *IEEE Trans. Mobile Comput.*, vol. 12, no. 6, pp. 1054-1064, 2013.
- [11] J. Su, Z. Sheng, L. Xie, G. Li, and A. X. Liu, "Fast splitting based tag identification algorithm for anti-collision in UHF RFID system," *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 2526-2538, 2019.
- [12] H. Guo, C. Hen, N. Wang, and M. Bolic, "PSR: a novel high-efficiency and easy-to-implement parallel algorithm for anticollision in RFID systems," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 1134-1145, 2016.
- [13] L. Zhang, W. Xiang, X. Tang, and Q. Yan, "A time-and energy-aware collision tree protocol for efficient large-scale RFID tag identification," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2406-2417, 2018.
- [14] Y. Lai, L. Hsiao, H. Chen, and J. Lin, "A novel query tree protocol with bit tracking in RFID tag identification," *IEEE Trans. Mobile Comput.*, vol. 12, no. 10, pp. 2063-2075, 2013.
- [15] L. Zhu and T.-S. Yum, "The optimal reading strategy for EPC Gen-2 RFID anti-collision systems," *IEEE Trans. Commun.*, vol. 58, no. 9, pp. 2725-2733, 2010.
- [16] W.-T. Chen, "An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 9-15, 2009.
- [17] V.-A. Javier, B.-D. Victoria, E.-L. Esteban, J. G.-C. Francisco, and J. Alcaraz, "Multiframe maximum-likelihood tag estimation for RFID anti-collision protocols," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 487-496, 2011.
- [18] B. Knerr, M. Holzer, C. Angerer, and M. Rupp, "Slot-wise maximum likelihood estimation of the tag population size in FSA protocols," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 578-585, 2010.
- [19] T. F. L. Porta, G. Maselli, and C. Petrioli, "Anti-collision protocols for single-reader RFID systems: temporal analysis and optimization," *IEEE Trans. Mobile Comput.*, vol. 10, no. 2, pp. 267-279, 2010.
- [20] P. Solic, J. Maras, and J. Radic, and Z. Blazevic, "Comparing theoretical and experimental results in Gen2 RFID throughput," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 349-357, 2017.
- [21] D.K. Klair, K.W. Chin, R. Raad, "A survey and tutorial of RFID anti-collision protocols," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 3, pp. 400-421, 2010.
- [22] W. Su, N.V. Alchazidis, T.T. Ha, "Multiple RFID tags access algorithm," *IEEE Trans. Mobile Comput.*, vol. 9, no. 2, pp. 174-187, 2010.
- [23] L. Zhang, W. Xiang, and X. Tang, "An efficient bit-detecting protocol for continuous tag recognition in mobile RFID systems," *IEEE Trans. Mobile Comput.*, vol. 17, no. 3, pp. 503-516, 2018.
- [24] V. Nambodiri and L. Gao, "Energy-aware tag anticollision protocols for RFID," *IEEE Trans. Mobile Comput.*, vol. 9, no. 1, pp. 44-59, 2010.
- [25] M. Shahzad and A. X. Liu, "Probabilistic optimal tree hopping for RFID identification," *IEEE/ACM Trans. Netw.*, vol. 23, no. 3, pp. 796-809, 2015.
- [26] L. Pan and H. Wu, "Smart trend-traversal protocol for RFID tag arbitration," *IEEE Trans. Wireless Commun.*, vol. 10, no. 11, pp. 3565-3569, 2011.
- [27] L. Zhang, J. Zhang, and X. Tang, "Assigned tree slotted Aloha RFID tag anti-collision protocols," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5493-5505, 2013.
- [28] X. Jia, Q. Feng, and L. Yu, "Stability analysis of an efficient anticollision protocol for RFID Tag Identification," *IEEE Trans. Commun.*, vol. 60, no. 8, pp. 2285-2294, 2012.
- [29] Y. Lai, L. Hsiao, and B. Lin, "Optimal slot assignment for binary tracking tree protocol in RFID tag identification," *IEEE/ACM Trans. Netw.*, vol. 23, no. 1, pp. 255-268, 2015.
- [30] L. Zhang, J. Zhang, and X. Tang, "An adaptive anti-collision protocol for large-scale RFID tag identification," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 601-604, 2014.
- [31] J. Su, Z. Sheng, G. Wen, and V. Leung, "A time efficient tag identification algorithm using dual prefix probe scheme (DPPS)," *IEEE Signal Process. Lett.*, vol. 23, no. 3, pp. 386-389, 2016.
- [32] J. Shin, B. Jeon, and D. Yang, "Multiple RFID tags identification with M-ary query tree scheme," *IEEE Wireless Commun. Lett.*, vol. 17, no. 3, pp. 604-607, 2013.
- [33] L. Zhang, W. Xiang, and X. Tang, "An efficient bit-detecting protocol for continuous tag recognition in mobile RFID system," *IEEE Trans. Mobile Comput.*, vol. 17, no. 3, pp. 503-516, 2018.
- [34] H. Landaluce, A. Perallos, E. Onieva, L. Arjona, and L. Bengtsson, "An energy and identification time decreasing procedure for memoryless RFID tag anti-collision protocols," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 4234-4247, 2016.
- [35] K. Finkenzeller, RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification, *John Wiley and Sons*, 2003.
- [36] J. Su, Y. Chen, Z. Sheng, and L. Sun, "Bit query based M-ary tree protocol for RFID tags identification," in *Proc. IEEE Global Commun. Conf. (GLOBECOM 2017)*, Singapore, 2017, pp. 1-7.
- [37] H. Yue, C. Zhang, M. Pan, Y. Fang, and S. Chen, "A time-efficient information collection protocol for large-scale RFID systems," in *Proc. ACM SIGCOMM 2012*, 2012, pp. 61-72.
- [38] J. Wang, H. Hassanieh, D. Katabi, and P. Indyk, "Efficient and reliable low-power backscatter networks," in *Proc. IEEE Int'l Conf. Comput. Commun. (INFOCOM 2012)*, 2012, pp. 2158-2166.
- [39] K. S. Leong, M. L. Ng, A. R. Grasso, and P. H. Cole, "Synchronization of RFID readers for dense RFID reader environment," in *Proc. Int'l Symp. App. Int. Workshops (SAINTW 2006)*, 2016, pp. 23-27.
- [40] C. Angerer and M. Rupp, "Advanced synchronization and decoding in RFID reader receivers," in *Proc. IEEE Radio and Wireless Symp.*, 2009, pp. 59-62.
- [41] X. Liu, X. Xie, S. Wang, D. Yao, J. Cao and K. Li, "Efficient range queries for large-scale sensor-augmented RFID systems," *IEEE/ACM Trans. Netw.*, vol. 27, no. 5, pp. 1873-1886, 2019.
- [42] M. Bonuccelli, F. Lonetti, and F. Martelli, "Instant collision resolution for tag identification in RFID networks," *Ad Hoc Netw.*, vol. 5, no. 8, pp. 1220-1232, 2007.
- [43] C. He, Z. Wang, and C. Miao, "Block-level unitary query: enabling orthogonal-like space-time code with query diversity for MIMO backscatter RFID," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1937-1949, 2016.



**Jian Su** has been a lecturer in the School of Computer and Software at the Nanjing University of Information Science and Technology since 2017. He received his PhD with distinction in communication and information systems at University of Electronic Science and Technology of China in 2016. He holds a B.S. in Electronic and information engineering from Hankou university and an M.S. in electronic circuit and system from Central China Normal University. His current research interests cover Internet of Things, RFID, and Wireless sensors networking.





**Yongrui Chen** has been an associate professor in the Department of Electronic, Electrical and Communication Engineering at the University of Chinese Academy of Sciences (UCAS) since 2014. He received his Ph.D. at UCAS in 2011, his M.S. at Tsinghua University in 2007, and his B.Sc. from Yanshan University in 2001. His current research interests cover the Internet of Things (IoT) and heterogeneous wireless networks.



**Zhengguo Sheng** has been a senior lecturer in the Department of Engineering and Design at the University of Sussex since 2015. He received his Ph.D. and M.S. with distinction at Imperial College London in 2011 and 2007, respectively, and his B.Sc. from the University of Electronic Science and Technology of China (UESTC) in 2006. His current research interests cover the Internet of Things (IoT), connected vehicles, and cloud/ edge computing.



**Zhong Huang** received B.S. and M.S. degrees from Henan University, Kaifeng, China, in 2011 and 2014, respectively. He is currently a research assistant working toward a Ph.D. degree at the School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests lie in dynamic spectrum access and cognitive radio-based networks, vehicular ad hoc networks, and wireless networks security.



**Alex X. Liu** received his Ph.D. degree in Computer Science from The University of Texas at Austin in 2006, and is a professor at the Department of Computer Science and Engineering, Michigan State University. He received the IEEE & IFIP William C. Carter Award in 2004, a National Science Foundation CAREER award in 2009, and the Michigan State University Withrow Distinguished Scholar Award in 2011. He has served as an Editor for IEEE/ACM Transactions on Networking, and he is currently an Associate Editor for IEEE Transactions

on Dependable and Secure Computing and IEEE Transactions on Mobile Computing, and an Area Editor for Computer Communications. He has served as the TPC Co-Chair for ICNP 2014 and IFIP Networking 2019. He received Best Paper Awards from ICNP-2012, SRDS-2012, and LISA-2010. His research interests focus on networking and security. He is a Fellow of the IEEE.